09/752,750

Amendments to the Claims

Claims 1-6. (canceled)

Claim 7. (currently amended) A computing system, comprising:

a rounding apparatus for rounding an input value to a nearest integer to accept an input value that is a real number represented in floating-point format, and to perform a rounding operation on the input value to generate an output value that is an integer represented in floating-point format, the rounding apparatus comprising:

a sign bit extractor to determine an extracted sign bit of the input value,

an adjustment generator to perform an "OR" operation on the extracted sign bit and a selected real value, generating an adjustment value,

an adding unit to compute an adjusted input value by adding the adjustment value to the input value, the adjusted input value being a real number represented in floating-point format,

a floating-point to integer converter to truncate a fractional portion of the adjusted input value to convert the adjusted input value to an integer represented in an integer format, and

an integer to floating-point converter to convert the integer represented in an integer format to generate the output value;

a memory to store a computer program that utilizes the rounding apparatus; and

a central processing unit (CPU) to execute the computer program, the CPU is cooperatively connected to the rounding apparatus and the memory,

~~The system of claim 5,~~

wherein the ~~"OR" operator~~ adjustment generator generates the adjustment value by performing a bit-wise logical OR operation on the sign bit and the selected ~~a~~ real value, where the selected real value ~~of~~ is 0.5.

Claim 8. (canceled)

Claim 9. (currently amended) A computing system, comprising:

a rounding apparatus for rounding an input value toward minus infinity (-∞) to accept an input value that is a real number represented in floating-point format, and to perform a rounding operation on the input value to generate an output value that is an integer represented in floating-point format, the rounding apparatus comprising:

~~The system of claim 8, wherein the rounding apparatus includes:~~

a floating-point to integer converter to truncate an input value to convert the input value to a first integer represented in an integer format;

an integer to floating-point converter to convert the first integer represented in an integer format to a second integer represented in floating-point format;

a ~~first SUBTRACT operator~~ portion generator to compute a fractional portion of the input value by subtracting ~~using~~ the second integer from the input value, generating a fractional portion of the input value;

a ~~"less than"~~ comparator to generate a boolean mask by comparing ~~based on~~ the fractional portion of the input value to a first selected real value;

an ~~"AND" operator~~ adjustment generator to generate an adjustment value represented in floating-point form using ~~to use~~ the boolean mask added to a second selected real value ~~to generate an adjustment value represented in floating-point format~~; and

a ~~second SUBTRACT operator~~ subtractor unit to generate an output value by subtracting ~~to subtract~~ the adjustment value from the input value ~~to generate the output value~~;

a memory to store a computer program that utilizes the rounding apparatus; and

a central processing unit (CPU) to execute the computer program, the CPU is cooperatively connected to the rounding apparatus and the memory.

Claim 10. (currently amended) The system of claim 9, wherein the ~~first SUBTRACT operator~~ portion generator computes the fractional portion of the input value by subtracting the second integer from the input value.

Claim 11. (currently amended) The system of claim 9, wherein the first selected real value is 0.0, and wherein the ~~"less-than"~~ comparator uses a "less than" compare to generates the boolean mask by comparing the fractional portion of the input value to the first selected ~~a~~ real value of 0.0.

Claim 12. (currently amended) The system of claim 9, wherein the second selected real value is 1.0, and wherein the adjustment generator ~~"AND" operator~~ generates the adjustment value by performing a bit-wise logical AND operation on the boolean mask and the second selected ~~a~~ real value of 1.0.

Claim 13. (canceled)

Claim 14. (currently amended) A computing system, comprising:

a rounding apparatus for rounding an input value toward plus infinity (+∞) to accept an input value that is a real number represented in floating-point format, and to perform a rounding operation on the input value to generate an output value that is an integer represented in floating-point format, the rounding apparatus comprising:

~~The system of claim 13, wherein the rounding apparatus includes:~~

a floating-point to integer converter to truncate an input value to convert the input value to a first integer represented in an integer format;

an integer to floating-point converter to convert the first integer represented in an integer format to a second integer represented in floating-point format;

a ~~SUBTRACT operator~~ portion generator to compute a fractional portion of the input value by subtracting ~~using~~ the second integer from the input value, generating a fractional portion of the input value;

a ~~"greater-than"~~ comparator to generate a boolean mask by comparing ~~based on~~ the fractional portion of the input value to a first selected real value;

an ~~"AND" operator~~ adjustment generator to generate an adjustment value represented in floating-point form using ~~to use~~ the boolean mask and a second selected real value ~~to generate an adjustment value represented in floating-point format~~; and

an ~~ADD operator~~ adding unit to generate an output value by adding ~~to add~~ the adjustment value to the input value ~~to generate the output value;~~

a memory to store a computer program that utilizes the rounding apparatus; and

a central processing unit (CPU) to execute the computer program, the CPU is cooperatively connected to the rounding apparatus and the memory.

Claim 15. (currently amended) The system of claim 14, wherein the ~~SUBTRACT operator~~ portion generator computes the fractional portion of the input value by subtracting the second integer from the input value.

Claim 16. (currently amended) The system of claim 14, wherein the first selected real value is 0.0, and wherein the ~~"greater than"~~ comparator generates the boolean mask by comparing the fractional portion of the input value to the first selected ~~a~~ real value of 0.0.

Claim 17. (currently amended) The system of claim 14, wherein the second selected real value is 1.0, and wherein the ~~"AND" operator~~ adjustment generator generates the adjustment value by performing a bit-wise logical AND operation on the boolean mask and the second selected ~~a~~ real value of 1.0.

Claims 18-22. (canceled)

Claim 23. (currently amended) ~~The method of claim 21~~ A method comprising:

building an adjustment value represented in floating-point format;

adding the adjustment value to an input value to generate an adjusted input value represented in floating-point format;

truncating the adjusted input value to convert the adjusted input value to a first integer represented in an integer format;

converting the first integer represented in an integer format to a second integer represented in floating-point format; and

storing the second integer as an output value,

wherein building the adjustment value comprises:

building the adjustment value by performing a bit-wise logical OR operation on a real value of 0.5 and a sign bit extracted from the input value.

Claim 24. (previously amended) The method as recited by claim 43, wherein additively combining the adjustment value with the input value comprises subtracting the adjustment value from the input value.

Claim 25. (original) The method of claim 24, wherein computing the fractional portion of the input value comprises:

subtracting the second integer represented in floating-point format from the input value to generate the fractional portion of the input value.

Claim 26. (original) The method of claim 24, wherein generating the boolean value comprises comparing the fractional portion of the input value to a real value of 0.0.

Claim 27. (original) The method of claim 24, wherein creating an adjustment value comprises performing a bit-wise logical AND operation on the boolean value and a real value of 1.0.

Claim 28. (previously amended) The method as recited in claim 43, wherein additively combining the adjustment value with the input value comprises

adding the adjustment value to the input value to generate a rounded input value, and wherein computing a fractional portion of the input values using the second integer represented

in floating-point format comprises subtracting the second integer represented in floating-point format from the input value.

Claim 29. (original) The method of claim 28, wherein creating an adjustment value comprises:

comparing the fractional portion of the input value to a real value of 0.0.

Claim 30. (original) The method of claim 28, wherein creating an adjustment value comprises:

performing a bit-wise logical AND operation on the boolean value and a real value of 1.0.

Claims 31-34. (canceled)

Claim 35. (currently amended) ~~The machine-readable medium of claim 34~~ A machine-readable medium comprising instructions which, when executed by a machine, cause the machine to perform operations comprising:

a first code segment to extract a sign bit of the input value;

a second code segment to generate an adjustment value based on the sign bit;

a third code segment to compute an adjusted input value represented in floating-point format;

a fourth code segment to truncate a fractional portion of the adjusted input value to convert the adjusted input value to an integer represented in an integer format; and

a fifth code segment to convert the integer represented in an integer format to generate the output value,

wherein the second code segment generates the adjustment value by performing a bit-wise logical OR operation on the sign bit and a value of 0.5.

Claims 36-39. (canceled)

Claim 40. (currently amended) A machine-readable medium comprising instructions which, when executed by a machine, cause the machine to perform operations comprising:

a first code segment to truncate an input value to convert the input value to a first integer represented in an integer format;

a second code segment to floating-point converter to convert the first integer represented in an integer format to a second integer represented in floating-point format;

a third code segment to subtract the second integer from the input value to compute a fractional portion of the input value;

a fourth code segment to generate a boolean mask based on the fractional portion of the input value;

a fifth code segment to generate an adjustment value represented in floating-point format by performing a bit-wise logical AND operation on the boolean mask and a selected real value; and

a sixth code segment to subtract the adjustment value from the input value to generate the output value represented in floating-point format.

Claim 41. (original) The machine-readable medium of claim 40, wherein the fourth code segment generates the boolean mask by comparing the fractional portion of the input value to a real value of 0.0.

Claim 42. (currently amended) The machine-readable medium of claim 40, wherein ~~the fifth code segment generates the adjustment value by performing a bit-wise logical AND operation on the boolean mask and a~~ selected real value ~~of~~ is 1.0.

Claim 43. (previously added) A method comprising:

generating a first integer represented in an integer format by truncating an input value;

converting the first integer represented in an integer format to a second integer represented in floating-point format;

computing a fractional portion of the input value using the second integer represented in floating-point format;

generating a boolean value using the fractional portion of the input value;

creating an adjustment value using the boolean value; and

computing a rounded input value by additively combining the adjustment value with the input value.

Claim 44. (previously added) The machine-readable medium as recited in claim 41, wherein the comparison of the fourth code segment generates a true Boolean mask when the fractional portion is less than the real value of 0.0.

Claim 45. (previously added) The machine-readable medium as recited in claim 41, wherein the comparison of the fourth code segment generates a true Boolean mask when the fractional portion is greater than the real value of 0.0.

Claim 46. (newly added) The method as recited in claim 43, wherein the generating a Boolean value generates a true Boolean mask when the fractional portion is less than a real value of 0.0.

Claim 47. (newly added) The method as recited in claim 43, wherein the generating a Boolean value generates a true Boolean mask when the fractional portion is greater than a real value of 0.0.

Claim 48. (newly added) A computing system, comprising:

a rounding apparatus to accept an input value that is a real number represented in floating-point format, and to perform a rounding operation on the input value to generate an output value that is an integer represented in floating-point format, the rounding apparatus comprising:

a floating-point to integer converter to truncate an input value to convert the input value to a first integer represented in an integer format,

an integer to floating-point converter to convert the first integer represented in an integer format to a second integer represented in floating-point format,

a portion generator to compute a fractional portion of the input value by subtracting the second integer from the input value, generating a fractional portion of the input value,

a comparator to generate a boolean mask by comparing the fractional portion of the input value to a first selected real value,

an adjustment generator to generate an adjustment value represented in floating-point form using the boolean mask added to a second selected real value, and

a subtractor unit to generate an output value by subtracting the adjustment value from the input value;

a memory to store a computer program that utilizes the rounding apparatus; and

a central processing unit (CPU) to execute the computer program, the CPU is cooperatively connected to the rounding apparatus and the memory.